

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

DOCKET: FIS920030157US1

PATENT

In re

Application of: Haswell et al.

) Examiner: M.M. Chaudry

)

Serial No.: 10/604,141

) Group Art Unit: 2133

)

Filing Date June 27, 2003

) Date: January 30, 2007

)

Title

**Method and System for Optimized Instruction Fetch to Protect  
Against Soft and Hard Errors**

Mail Stop Appeal Brief - Patents  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**BRIEF FOR APPELLANTS**

This is an appeal from the final rejection by the Examiner mailed September 19, 2006, rejecting claims 1-20. A notice of appeal and the appeal fee were timely mailed on December 19, 2006. Please charge the \$500.00 fee for this appeal brief to Deposit Account No. 09-0458 of the assignee, International Business Machines Corporation.

### **REAL PARTY IN INTEREST**

The real party in interest is the assignee of all rights in this application, International Business Machines Corporation, a corporation of the State of New York, having a place of business at Armonk, New York.

### **RELATED APPEALS AND INTERFERENCES**

There are no appeals or interferences known to appellants, appellants' legal representatives or assignee, which will directly affect or be affected by, or have a bearing on the Board's decision on this appeal.

### **STATUS OF CLAIMS**

The subject application was filed on June 27, 2003 with claims 1-20. An amendment was filed on December 15, 2005, responsive to the office action mailed September 19, 2005, amending claims 1, 2, 11, 16 and 17. In an office action mailed September 19, 2006, a final rejection was made of all of the claims in the application, to wit, claims 1-20. Appellants are appealing the rejection of these claims.

### **STATUS OF AMENDMENTS**

All the amendments made during prosecution of the application have been entered and are presently in the application. The rejected claims 1-20 as they presently stand are set forth in the Appendix. A summary of the rejection of the claims may be found in the Office Action mailed September 19, 2006.

### **SUMMARY OF CLAIMED SUBJECT MATTER**

Appellants' invention is directed to a method of detecting error during transfer of data signals from a data memory to a computer processor (claim 1), a system for detecting

error during transfer of data signals from a data memory to a computer processor (claim 11) and a program storage device embodying executable instructions to perform method steps for detecting error during transfer of a raw data signal and an error detection code for the raw data signal from a data memory to a computer processor, wherein the raw data signal includes an error detection code (claim 16).

As defined in independent claim 1, the method of detecting error during transfer of data signals from a data memory 24 (Fig. 2) to a computer processor 12 (Figs. 1 and 2) begins with the commencement of transmission of a raw data signal 29 (Fig. 2) and an error detection code for the raw data signal 29 (Fig. 2) from a data memory 24 (Fig. 2) to a computer processor 12 (Figs. 1 and 2). Specification, ¶0010, ll.1-6, ¶0028, ll.4-7, ¶0029, ll.4-13, ¶0030, ll.1-8 and ¶¶0052, ll.1-8. At the time of the commencement of transmission of the raw data signal 29 (Fig. 2) from the data memory 24 (Fig. 2) to the computer processor 12 (Figs. 1 and 2), the raw data signal 29 (Fig. 2) is checked for corruption based on its error detection code (from error correcting code, generation checking unit 26, Fig. 2). Specification, ¶0010, ll.6-8 and ¶0031, ll.1-3. If the data has not been corrupted, the transmission of the raw data signal 29 (Fig. 2) to the computer processor 12 (Figs. 1 and 2) is completed. Specification, ¶0010, ll.8-10 and ¶0029, ll.4-6. If the error detection code indicates data corruption, the raw data signal 29 (Fig. 2) is substituted with a predetermined reserved data signal 30 (Fig. 2) which is then transmitted to the computer processor 12 (Figs. 1 and 2). Specification, ¶0010, ll.10-13, and ¶0030, ll.3-8. The computer processor 12 (Figs. 1 and 2) then processes the data signal. Specification, ¶0010, ll.13-14, ¶0029, ll.4-6, ¶0031, ll.1-3 and ¶¶0032-0035.

A system for detecting error during transfer of data signals from a data memory 24 (Fig. 2) to a computer processor 12 (Figs. 1 and 2) is defined in independent claim 11. The system includes a computer processor 12 (Figs. 1 and 2) and a data memory device 24 (Fig. 2) containing raw data 29 (Fig. 2) to be processed by the computer processor 12 (Figs. 1 and 2), wherein the raw data 29 (Fig. 2) is in the form of a raw data signal 29 (Fig. 2) including an error detection code (from error correcting code, generation checking unit 26, Fig. 2). Specification, ¶0018, ll.1-5, ¶0028, ll.4-7, ¶0029, ll.4-13, ¶0030, ll.1-8, ¶¶0052, ll.1-8. and ¶¶0052, ll.1-8. The system further includes a data checker device 26 (Fig. 2) adapted to check the raw data 29 (Fig. 2) for presence of data corruption and, if the error detection code indicates data corruption, determining if the corrupted data in the original raw data 29 (Fig. 2) may be corrected. Specification, ¶0018, ll.5-8, ¶0029, ll.8-11 and ¶0030, ll.8-10. The system also includes a computer processor instruction unit 10 (Figs. 1 and 2). Specification, ¶0018, ll.8-9, ¶0027, ll.1-6 and ¶0028, ll.1-7. The instruction unit 10 (Figs. 1 and 2) is adapted to commence transmission of a raw data 29 (Fig. 2) from the data memory device 24 (Fig. 2) to the computer processor 12 (Figs. 1 and 2) and cause the data checker device 26 (Fig. 2) to check the raw data signal 29 (Fig. 2) for corruption via decode of the data and error detection code. Specification, ¶0018, ll.9-12, ¶0027, ll.1-6 and ¶0029, ll.6-11. The instruction unit 10 (Figs. 1 and 2) is further adapted to cause completion of transmission of the raw data 29 (Fig. 2) to the computer processor 12 (Figs. 1 and 2) if the error detection code indicates no data corruption or, if the error detection code indicates data corruption, cause substitution of the raw data 29 (Fig. 2) with a predetermined reserved instruction 30 (Fig. 2) and transmission of the predetermined

reserved instruction 30 (Fig. 2) to the computer processor 12 (Figs. 1 and 2). Specification, ¶0018, ll.12-16, ¶0030, ll.1-12 and ¶¶0031-0035.

Independent claim 16 is directed to a program storage device 10 (Figs. 1 and 2) readable by a machine 12 (Figs. 1 and 2), tangibly embodying a program of instructions executable by the machine 12 (Figs. 1 and 2) to perform method steps for detecting error during transfer of raw data signals 29 (Fig. 2) from a data memory 24 (Fig. 2) to a computer processor 12 (Figs. 1 and 2), wherein the raw data signal 29 (Fig. 2) includes an error detection code (from error correcting code, generation checking unit 26, Fig. 2). Specification, ¶0017, ll.1-4 and ¶0027, ll.1-5. These method steps include, at the time of the commencement of transmission of the raw data signal 29 (Fig. 2) from the data memory 24 (Fig. 2) to the computer processor 12 (Figs. 1 and 2), checking the raw data signal 29 (Fig. 2) for corruption. Specification, ¶0010, ll.6-8 and ¶0031, ll.1-3. If the error detection code indicates no data corruption, the transmission of the raw data signal 29 (Fig. 2) to the computer processor 12 (Figs. 1 and 2) is completed. Specification, ¶0010, ll.8-10 and ¶0029, ll.4-6. If the error detection code indicates data corruption, the raw data signal 29 (Fig. 2) is substituted with a predetermined reserved data signal 30 (Fig. 2) which is then transmitted to the computer processor 12 (Figs. 1 and 2). Specification, ¶0010, ll.10-13, and ¶0030, ll.3-8. The computer processor 12 (Figs. 1 and 2) then processes the data signal. Specification, ¶0010, ll.13-14, ¶0029, ll.4-6, ¶0031, ll.1-3 and ¶¶0032-0035.

If the error detection code indicates data corruption, claims 2 and 17 specify further determining if the corrupted data in the original raw data signal 29 (Fig. 2) may be

corrected, and subsequently retrieving the corrected data 31 (Fig. 2) and processing the corrected data signal 31 (Fig. 2) with the computer processor 12 (Figs. 1 and 2). Specification, ¶0010, ll.14-18, ¶0030, ll. 8-12 and ¶¶0032-0035.

Claims 3 and 18 state that if the computer processor 12 (Figs. 1 and 2) processes the predetermined reserved signal 30 (Fig. 2), the method may also include determining whether corrupted data in the raw data signal 29 (Fig. 2) has been corrected, and, if corrected, subsequently retrieving the corrected data 31 (Fig. 2) and processing the corrected data signal 31 (Fig. 2) with the computer processor 12 (Figs. 1 and 2). Specification, ¶0011, ll.1-4, ¶0030, ll. 8-12 and ¶ ¶0031-0035.

Claims 5 and 12 specify that if the computer processor 12 (Figs. 1 and 2) processes a predetermined reserved instruction 30 (Fig. 2), the computer processor 12 (Figs. 1 and 2) executes an error handling routine comprising determining whether corrupted data in the raw data signal 29 (Fig. 2) has been corrected, and, if corrected, retrieving the corrected data signal 31 (Fig. 2), processing the corrected data signal 31 (Fig. 2) with the computer processor 12 (Figs. 1 and 2), and replacing the raw data signal 29 (Fig. 2) in the data memory 24 (Fig. 2) with the corrected data signal 31 (Fig. 2). Specification, ¶0012, ll.1-6, ¶0019, ll.1-5 and ¶¶0031-0035.

In claims 6 and 13, it is stated that the computer processor 12 (Figs. 1 and 2) operates on timed, uniform clock 16 (Fig. 1) cycles, and the invention includes transmission of the raw data signal 29 (Fig. 2) from the data memory 24 (Fig. 2) to a computer processor 12 (Figs. 1 and 2), simultaneously checking of the raw data signal 29 (Fig. 2) for corruption, and transmission of either the raw data signal 29 (Fig. 2) or

predetermined reserved signal 30 (Fig. 2) to the computer processor 12 (Figs. 1 and 2), within a single clock 16 (Fig. 1) cycle. Specification, ¶0013, ll.1-5, ¶0020, ll.1-7, ¶0027, ll.5-8 and ¶0030, ll.1-8.

### **GROUND OF REJECTION TO BE REVIEWED ON APPEAL**

The contested issues in this appeal are whether claims 1-20 are obvious to one of ordinary skill in the art under 35 USC § 103(a) from Sakamoto U.S. Patent No. 4,617,660 in view of Albonesi U.S. Patent No. 4,920,539.

### **ARGUMENT**

#### **I. Prior Art**

U.S. Patent No. 4,617,660 to Sakamoto is directed to a faulty-memory processing method and apparatus in a data processing system which executes a time-sharing data process with breaks. Hard error which may exist in a cell in a normal memory is detected by using an error correction circuit. After correcting an error in information read out from a detected hard-error cell in the normal memory, information including the above corrected information with respect to the detected hard-error cell is transcribed into a relief memory. The above correction and transcription is executed during the breaks in the time-sharing data process.

The Examiner has admitted, "Sakamoto does not explicitly teach to check the data signal for corruption at the time it is received by the computer processor as stated in the present application." September 19, 2006 Office Action, p.5. However, not only does Sakamoto not teach this, he explicitly teaches against checking the data signal for corruption at the time it is received by the computer processor:

*Under a normal operation mode, the CPU first receives read data from the MU and executes a process using the received read data before detection and correction of an error in the read data, without waiting for the result of the error detection. When an error is detected in the read data, the process by the CPU is stopped in response to a memory error signal S<sub>b</sub>, and then corrected data is written back in the memory. Thereafter, the read-out operation from the memory is retried. Thus, the access time can be shortened, and high speed processing can be executed.*

Sakamoto, column 4, lines 31-41 (emphasis added). Thus, it is clear that Sakamoto teaches that the processor should execute the process using the read data before detecting and correcting any error in that data. The above passage also teaches that recovery from an error involves stopping the processor, and retrying once the data has been corrected.

U.S. Patent No. 4,920,539 to Albonesi is directed to a system for correcting soft memory failures such as alpha particle failures in a dynamic random access memory and in a computer system wherein writeback caches are employed in a system bus environment. The address field and source identification code associated with a detected data error are stored. A generic bus request signal is generated and upon a bus grant a read message is issued on the system bus having an address field and destination address code corresponding to the stored address field and source identification code. In response to the read message, the device indicated by the identification code writes back to memory the correct data corresponding to the address field.

Albonesi states, "it is an object of the present invention to provide an improved system for memory error correction and, in particular, for correction of alpha particle type (soft) memory failures. Another object of the present invention is to provide a memory



error correction system for use in a system wherein the various processors employ writeback caches." Albonesi, column 1, lines 57-64.

In the only portion of Albonesi cited against independent claims 1, 11 and 16 of the instant application, he states the following in claim 1:

In a computer system having a processing unit, memory, memory control unit, system communication bus, and bus control unit, a method of correcting memory errors, comprising the steps of, detecting a data error while data is being transferred from memory to the system bus and generating corresponding corrected data if a data error is detected, storing at the memory control unit at least the address field and source identification code associated with the just detected data error, generating a bus request signal coupled to the bus control unit, said bus control unit, in turn, generating a bus grant signal, said memory control unit in response to said bus grant signal issuing a read message on the system bus having an address field and destination identification code corresponding to said stored address field and source identification code, and in response to said read message, the device indicated by said identification code, writing back to memory the correct data corresponding to said address field.

Albonesi, column 8, lines 11-29.

Albonesi does not disclose or suggest substituting the raw data signal with a predetermined reserved data signal in the event of an error, and transmitting the predetermined reserved signal to a computer processor.

## **II. The Examiner's Rejections and Appellants' Arguments as to Non-Obviousness**

### **A. Claims 1, 11 and 16**

The prior art cited by the Examiner fails to teach in an error detecting method or system the combination of 1) checking the raw data signal for corruption at the time of the commencement of transmission of the raw data signal from the data memory to the computer processor, and 2) if the error detection code indicates data corruption, substituting the raw data signal with a predetermined reserved data signal and transmitting

the predetermined reserved signal to the computer processor. The Examiner has not made a *prima facie* case of obviousness with respect to these claims.

The Sakamoto patent teaches that the processor should execute the process using the read data before detecting and correcting any error in that data. This is substantially different from the claimed invention in that Sakamoto's method allows processing to proceed with faulty data for some time.

Sakamoto's approach can be particularly problematic when, for example, the data is actually an instruction and the CPU executes the wrong instruction or perhaps attempts to execute an invalid instruction. An invalid instruction might also cause the CPU to grind to a halt before it is stopped by the memory error signal. Sakamoto teaches that recovery from an error involves stopping the processor, and retrying once the data has been corrected. In order to do so, the system must have the capability to re-create the exact state of CPU operations prior to the error, even after it has proceeded for some time using corrupted data. Such a capability can be rather costly in terms of additional logic, memory, registers, and the like, and difficult to implement in today's pipelined processors. It is therefore a unique advantage of the present invention to swap in a predetermined reserved signal to the processor, as taught in claims 1, 11 and 16. The processor will then never be required to execute an invalid or incorrect instruction, nor will it be required to stop processing. Instead, it can jump to an error handling routine, which will cause the corrected data to be retrieved once it is available, and may also include the ability to process another task in the interim.

The Examiner has not taken issue with or contested the fact that Sakamoto's method teaches away from checking a data signal for corruption at the time it is received by the computer processor, as stated in the present application. More importantly, the Examiner has not provided a motivation that one of ordinary skill in the art would have to use an opposite approach, as the Examiner contends is taught by Albonesi.

The portion of the Albonesi patent that the Examiner cites for the error correction clearly is contrary to the teaching of Sakamoto. This is the passage at column 8, lines 11-29 referring to the correction of memory errors by detecting data error while the data is being transferred from memory to the system bus and generating corresponding corrected data if a data error is detected. The Examiner has not explained how or why one of ordinary skill in the art, reading the references as a whole, would abandon the clear directive of the primary reference, Sakamoto, that teaches away from checking a data signal for corruption at the time it is received by the computer processor, and use the opposite approach favored by Albonesi. See, *In re Lee*, 277 F.3d 1338, 61 USPQ2d 1430, 1433 (Fed. Cir. 2002) ("When patentability turns on the question of obviousness, the search for and analysis of the prior art includes evidence relevant to the finding of whether there is a teaching, motivation or suggestion to select and combine the references relied on as evidence of obviousness."); *McGinley v. Franklin Sports, Inc.*, 262 F.3d 1339, 60 USPQ2d 1001, 1008 (Fed. Cir. 2001) ("the central question is whether there is reason to combine [the] references") Appellants contend this is due entirely to improper hindsight analysis, using the teachings of appellants' own specification against them. See, *W.L. Gore*

*v. Carlock, Inc.*, 721 F.2d 1540, 220 USPQ 303, 312-13 (Fed. Cir. 1983) (It is improper to use "that which the inventor taught against the teacher.")

Moreover, the cited passage in the Albonesi patent is in the claims of the patent. Accordingly, one must read the Albonesi specification to understand what is meant by the cited claim passage. See *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 34 USPQ2d 1321, 1329 (Fed. Cir. 1995) (en banc), *aff'd*, 517 U.S. 370, 38 USPQ2d 1461 (1996) ("Claims must be read in view of specification, of which they are a part.") (citing *Autogiro Co. of America v. United States*, 384 F.2d 391, 397, 155 USPQ 697, 702 (Ct. Cl. 1967)).

With this in mind, a careful reading of the Albonesi specification makes it clear that Albonesi does not disclose or suggest "substituting the raw data signal with a predetermined reserved data signal and transmitting the predetermined reserved signal to the computer processor" as appellants claim. Albonesi states, "it is an object of the present invention to provide an improved system for memory error correction and, in particular, for correction of alpha particle type (soft) memory failures. Another object of the present invention is to provide a memory error correction system for use in a system wherein the various processors employ writeback caches." Albonesi, column 1, lines 57-64. In accordance with Albonesi's objects, what is described is basic, long-known error correction code (ECC) function, with a write-back of corrected data to memory. Therefore, in Albonesi's method, the destination of the data must wait for the correction to take place before receiving the data.

Albonesi simply does not address the problem solved by appellants' invention, whereby a processor cannot wait for the ECC logic to correct the error. If a fast processor were the destination in the Albonesi case, the processor would have to pause until the ECC correction was complete. This is referred to as "starving the processor" and is generally undesirable. Moreover, it is not what appellants claim in the instant invention. Accordingly, the broad interpretation by the Examiner made by reading only the cited claim passage in Albonesi is not supported or taught by a reading of the Albonesi patent as a whole.

The instant invention resolves this problem by reacting to the initial error detection signal, available before the corrected data. When an error is indicated, a predetermined reserved data signal is given to the processor in place of the corrupted data. This allows the processor to proceed with a valid instruction, even though there had been an error. In the meantime the data may be corrected and stored for subsequent processing. The claimed method and system using the instant invention therefore can take advantage of the fact that a valid instruction will always be given to the processor without an ECC delay (regardless of whether or not there was an error), and can operate at a higher performance level.

Accordingly, in order to combine the teachings of Sakamoto and Albonesi in the manner hypothesized, one of ordinary skill in the art would have to ignore the clear and direct teachings of Sakamoto that the processor "executes a process using the received read data before detection and correction of an error in the read data, without waiting for the result of the error detection." Sakamoto, column 4, lines 31-41. There is no motivation

given in either reference for proceeding against such a teaching. However, even if one did ignore Sakamoto's teaching in this regard, the Albonesi method would still not arrive at or suggest appellants' invention because the processor would have to wait for corrected data to proceed, and would not "substitut[e] the raw data signal with a predetermined reserved data signal and transmit[ ] the predetermined reserved signal to the computer processor" for processing as in appellants' invention. Thus, the hypothetical combination of Sakamoto and Albonesi does not render appellants' invention of claims 1, 11 and 16 *prima facie* obvious.

B. Claims 2, 3, 5, 12, 17 and 18

Appellants' dependent claims 2, 3, 5, 12, 17 and 18 recite that only after the reserved data signal has been processed does the data correction occur in appellants' invention. Since neither Sakamoto nor Albonesi disclose or suggest processing a reserved data signal, appellants' claims 2, 3, 5, 12, 17 and 18 are not obvious from these references.

C. Claims 6 and 13

In claims 6 and 13, appellants specify that the computer processor operates on timed, uniform clock cycles, and that the steps of transmitting the raw data signal from the data memory to a computer processor, simultaneously checking the raw data signal for data corruption, and transmitting either the raw or predetermined reserved signal to the computer processor are performed within a clock cycle. Such operations are nowhere disclosed or suggested in either Sakamoto or Albonesi. The Examiner's citation of a "refresh timing signal" in Sakamoto does not disclose appellants' performance within a

clock cycle of the steps of transmitting the raw data signal from the data memory to a computer processor, simultaneously checking the raw data signal for data corruption, and transmitting either the raw or predetermined reserved signal to the computer processor, as defined in claims 6 and 13.

### CONCLUSION

For the reasons given above, appellants submit that the instant application is not obvious from the cited prior art of Sakamoto U.S. Patent No. 4,617,660 in view of Albonesi U.S. Patent No. 4,920,539. Reversal of the rejection under 35 USC § 103(a) is respectfully requested.

Respectfully submitted,

A handwritten signature in black ink, appearing to read "Peter W. Peterson", written over a horizontal line.

Peter W. Peterson  
Reg. No. 31,867

**DeLIO & PETERSON, LLC**  
121 Whitney Avenue  
New Haven, CT 06510-1241  
(203) 787-0595

**CLAIMS APPENDIX****Rejected Claims of Serial No. 10/604,141**

1. (previously presented) A method of detecting error during transfer of data signals from a data memory to a computer processor comprising:

commencing transmission of a raw data signal and an error detection code for the

raw data signal from a data memory to a computer processor;

at the time of the commencement of transmission of the raw data signal from the

data memory to the computer processor, checking the raw data signal for corruption based on its error detection code;

if the data has not been corrupted, completing transmission of the raw data signal to the computer processor,

if the error detection code indicates data corruption, substituting the raw data signal

with a predetermined reserved data signal and transmitting the predetermined reserved signal to the computer processor;

processing the data signal with the computer processor.

2. (previously presented) The method of claim 1 wherein, if the error detection code indicates data corruption, further including determining if the corrupted data in the raw data signal may be corrected, and subsequently retrieving the corrected data and processing the corrected data signal with the computer processor.



3. (original) The method of claim 1 further including, if the computer processor processes the predetermined reserved signal, determining whether corrupted data in the raw data signal has been corrected, and, if corrected, retrieving the corrected data signal and processing the corrected data signal with the computer processor.

4. (original) The method of claim 1 further including, if the corrupted data in the raw data signal has been corrected, replacing the raw data signal in the data memory with the corrected raw data signal.

5. (original) The method of claim 1 further including, if the computer processor processes a predetermined reserved signal, the computer processor executes an error handling routine comprising determining whether corrupted data in the raw data signal has been corrected, and, if corrected, retrieving the corrected data signal, processing the corrected data signal with the computer processor, and replacing the raw data signal in the data memory with the corrected data signal.

6. (original) The method of claim 1 wherein the computer processor operates on timed, uniform clock cycles, and wherein the steps of transmitting the raw data signal from the data memory to a computer processor, simultaneously checking the raw data signal for data corruption, and transmitting either the raw or predetermined reserved signal to the computer processor are performed within a clock cycle.

7. (original) The method of claim 1 further including commencing transmission of a subsequent raw data signal from the data memory to the computer processor and repeating the aforementioned steps for the subsequent raw data signal, until all desired raw data signals from the data memory are processed by the computer processor.
8. (original) The method of claim 1 wherein the raw data signal is a multi-bit data signal.
9. (original) The method of claim 1 wherein checking of the raw data signal for corruption by decoding the data and error detection code is performed simultaneously with the commencement of transmission of the raw data signal from the data memory to the computer processor.
10. (original) The method of claim 1 wherein the raw data signal is an instruction from a random access memory associated with the computer processor.
11. (previously presented) A system for detecting error during transfer of data signals from a data memory to a computer processor comprising:
  - a computer processor;
  - a data memory device containing raw data and an error detection code for the raw data to be processed by the computer processor, the raw data being in the form of a raw data signal;

a data checker device adapted to check the raw data for corruption and, if the error detection code indicates data corruption, correcting the corrupted data in the raw data; and

a computer processor instruction unit, the instruction unit adapted to commence transmission of a raw data from the data memory device to the computer processor and cause the data checker device to check the raw data signal for corruption, the instruction unit further adapted to cause completion of transmission of the raw data to the computer processor if the error detection code indicates no data corruption or, if the error detection code indicates data corruption, cause substitution of the raw data with a predetermined reserved instruction and transmission of the predetermined reserved instruction to the computer processor.

12. (original) The system of claim 11 wherein the computer processor, upon processing a predetermined reserved instruction, is further adapted to execute an error handling routine comprising determining whether corrupted data in the raw data has been corrected, and, if corrected, retrieving the corrected data, processing the corrected data with the computer processor, and replacing the raw data in the data memory device with corrected data.

13. (original) The system of claim 11 wherein the computer processor includes a clock, the processor adapted to operate on timed, uniform clock cycles produced by the clock,

and wherein the computer processor instruction unit is adapted to cause transmission of the raw data from the data memory device to the computer processor, the data checker device is adapted to check simultaneously the raw data for presence of corruption, and the computer processor instruction unit is adapted to transmit either the raw data or predetermined reserved instruction to the computer processor, within a clock cycle.

14. (original) The system of claim 11 further including a corrected data register adapted to receive data corrected by the data checker device and transmit corrected data to the computer processor.

15. (original) The system of claim 11 wherein the data memory device is a random access memory associated with the computer processor and the raw data is an instruction from the random access memory for the computer processor.

16. (previously presented) A program storage device readable by a machine, embodying a program of instructions executable by the machine to perform method steps for detecting error during transfer of a raw data signal and an error detection code for the raw data signal from a data memory to a computer processor, the raw data signal including an error detection code, the method comprising:

at the time of the commencement of transmission of the raw data signal from the data memory to the computer processor, checking the raw data signal for corruption;

if the error detection code indicates no data corruption, completing transmission of the raw data signal to the computer processor,

if the error detection code indicates data corruption, substituting the raw data signal with a predetermined reserved data signal and transmitting the predetermined reserved signal to the computer processor;

processing the data signal with the computer processor.

17. (previously presented) The program storage device of claim 16 wherein, if the error detection code indicates data corruption, the method further includes determining if the corrupted data in the raw data signal may be corrected, and subsequently retrieving the corrected data and processing the corrected data signal with the computer processor.

18. (original) The program storage device of claim 16 wherein the method further includes, if the computer processor processes the predetermined reserved instruction, determining whether corrupted data in the raw data signal has been corrected, and, if corrected, retrieving the corrected data signal and processing the corrected data signal with the computer processor.

19. (original) The program storage device of claim 16 wherein the method further includes, if the corrupted data in the raw data signal has been corrected, replacing the raw data signal in the data memory with the corrected data signal.

20. (original) The program storage device of claim 16 wherein the error detection code comprises ECC code, and the raw data signal is an instruction from a random access memory associated with the computer processor.

EVIDENCE APPENDIX

None

**RELATED PROCEEDINGS APPENDIX**

None